**MODERN COLLEGE OF ARTS,SCI. & COMM. PUNE-05.**

**DEPARTMENT OF STATISTICS.(Autonomous)**

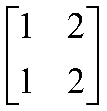
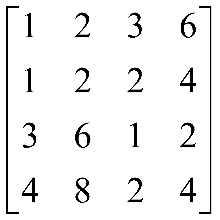
**M.Sc.( I )- ST-15**

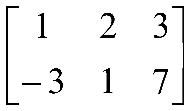
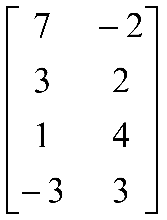
**EXPT.NO.  4**

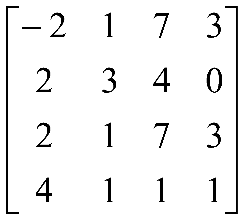
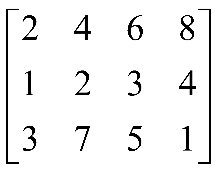
**Title : Inverse of  a square matrix (Direct method, g-inverse,  MP g-inverse , Partitioning method)**

Q1. Find g- inverse and MP g-inverse  of the given matrix in each of the following

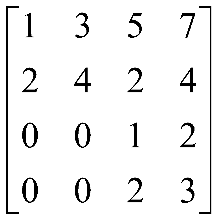
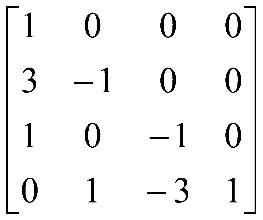
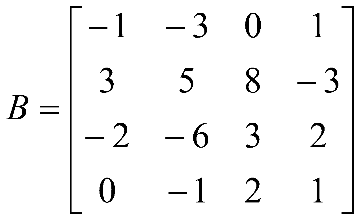


i)  A=ii)  B=

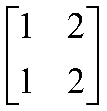
iii) A=  iv) A = 

v) A=  vi) D=

Q2. Obtain the inverse of the matrix B by partitioning method.

1. B=    ii) B=  iii)

**\*\*\*\*\***

**Q1.** i)  A=

A=[1 2;1 2]

A =

1 2

1 2

>> r=rank(A)

r = 1

>> a1=A(2,:)-A(1,:)

a1 =

1. 0

>> B=[1 2;a1]

B =

1 2

0 0

>> C=A(1,:)

C =

1 2

>> C=A(:,2)

C =

2

2

>> R=B(1,:)

R =

1 2

MP=R'\*inv(C'\*A\*R')\*C'

MP =

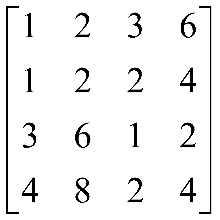
0.10000 0.10000

0.20000 0.20000

>> C\*R

ans =

2 4

**ii)** B=

>> B=[1 2 3 6;1 2 2 4;3 6 1 2;4 8 2 4]

B =

1 2 3 6

1 2 2 4

3 6 1 2

4 8 2 4

>> r=rank(B)

r = 2

>> b1=rref(B)

b1 =

1 2 0 0

0 0 1 2

0 0 0 0

0 0 0 0

>> C=[B(:,1),B(:,3)]

C =

1 3

1 2

3 1

4 2

C=[B(:,1),B(:,3)]

C =

1 3

1 2

3 1

4 2

>> R=[b1(1,:);b1(2,:)]

R =

1 2 0 0

0 0 1 2

>> MP=R'\*inv(C'\*B\*R')\*C'

MP =

-0.0260870 -0.0121739 0.0330435 0.0347826

-0.0521739 -0.0243478 0.0660870 0.0695652

0.0565217 0.0330435 -0.0182609 -0.0086957

0.1130435 0.0660870 -0.0365217 -0.0173913

>> C\*R

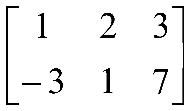
ans =

1 2 3 6

1 2 2 4

3 6 1 2

4 8 2 4

**iii).**A=  

>> A=[1 2 3;-3 1 7]

A =

1 2 3

-3 1 7

>> r=rank(A)

r = 2

>> a1=rref(A)

a1 =

1.00000 0.00000 -1.57143

0.00000 1.00000 2.28571

>> C=[A(,:1),A(,:2)]

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> C=[A(:,1),A(:,2)]

C =

1 2

-3 1

>> R=[a1(1,:);a1(2,:)]

R =

1.00000 0.00000 -1.57143

0.00000 1.00000 2.28571

>> MP=R'\*inv(C'\*A\*R')\*C'

MP =

0.279343 -0.145540

0.230047 -0.061033

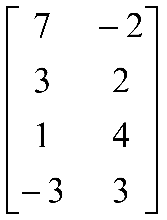
0.086854 0.089202

C\*R

ans =

1.0000 2.0000 3.0000

-3.0000 1.0000 7.0000

iv). A = 

A=[7 2;3 2;1 4;3 3]

A =

7 2

3 2

1 4

3 3

>> r=rank(A)

r = 2

>> a1=rref(A)

a1 =

1 0

0 1

0 0

0 0

>> C=[A(,:1),A(,:2)]

>> C=[A(:,1),A(:,2)]

C =

7 2

3 2

1 4

1. 3

>> R=[a1(1,:);a1(2,:)]

R =

1 0

0 1

>> MP=R'\*inv(C'\*A\*R')\*C'

MP =

0.14286 0.02857 -0.08571 0.00000

-0.08225 0.03203 0.20693 0.09091

>> C\*R

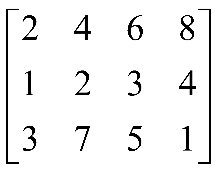
ans =

7 2

3 2

1 4

1. 3

**vi).** D=

A=[2 4 6 8;1 2 3 4;3 7 5 1]

A =

2 4 6 8

1 2 3 4

3 7 5 1

>> r=rank(A)

r = 2

>> a=rref(A)

a =

1.00000 0.00000 11.00000 26.00000

0.00000 1.00000 -4.00000 -11.00000

0.00000 0.00000 0.00000 0.00000

>> C=[A(:,1),A(:,2)]

C =

2 4

1 2

3 7

>> R=[a(1,:);a(2,:)]

R =

1.00000 0.00000 11.00000 26.00000

0.00000 1.00000 -4.00000 -11.00000

>> MP=R'\*inv(C'\*A\*R')\*C'

MP =

-0.0078431 -0.0039216 0.0441176

-0.0274510 -0.0137255 0.1127451

0.0235294 0.0117647 0.0343137

0.0980392 0.0490196 -0.0931373

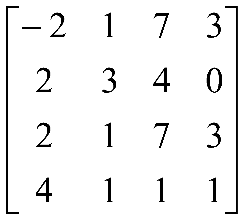
>> C\*R

ans =

2.0000 4.0000 6.0000 8.0000

1.0000 2.0000 3.0000 4.0000

3.0000 7.0000 5.0000 1.0000

**v).** A=  

A=[-2 1 7 3;2 3 4 0;2 1 7 3;4 1 1 1]

A =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> r=rank(A)

r = 4

>> a=rref(A)

a =

1 0 0 0

0 1 0 0

0 0 1 0

0 0 0 1

>> C=[A(:,1),A(:,2),A(:,3),A(:,4)]

C =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> R=[a(1,:);a(2,:);a(3,:);a(4,:)]

R =

1 0 0 0

0 1 0 0

0 0 1 0

0 0 0 1

C\*R

ans =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> MP=R'\*inv(C'\*A\*R')\*C'

MP =

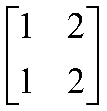
-2.5000e-01 3.3307e-16 2.5000e-01 -1.4433e-15

6.0000e-01 2.0000e-01 -8.0000e-01 6.0000e-01

-3.2500e-01 1.0000e-01 4.7500e-01 -4.5000e-01

7.2500e-01 -3.0000e-01 -6.7500e-01 8.5000e-01

**G inverse**

Q1. )  A=

>> A=[1 2; 1 2]

A =

1 2

1 2

>> r=rank(A)

r = 1

>> B=[2]

B = 2

>> r1=rank(B)

r1 = 1

>>Binv=inv(B)

Binv= 0.50000

>>BinvT=Binv'

BinvT= 0.50000

>> A1=[0 0.5;0 0]

A1 =

0.00000 0.50000

0.00000 0.00000

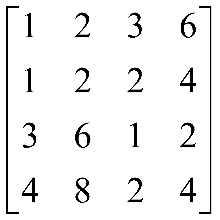
>> G=A1'

G =

0.00000 0.00000

0.50000 0.00000

>> #g is g-inverse

Q2. B=

B=[1 2 3 6;1 2 2 4;3 6 1 2;4 8 2 4]

B =

1 2 3 6

1 2 2 4

3 6 1 2

4 8 2 4

>> r=rank(B)

r = 2

>> C=[2 2;6 1]

C =

2 2

6 1

>> r1=rank(C)

r1 = 2

>>Cinv=inv(C)

Cinv =

-0.10000 0.20000

0.60000 -0.20000

>>CinvT=Cinv'

CinvT =

-0.10000 0.60000

0.20000 -0.20000

B1=[0 0 0 0;0 -0.1 0.6 0;0 0.2 -0.2 0;0 0 0 0]

B1 =

0.00000 0.00000 0.00000 0.00000

0.00000 -0.10000 0.60000 0.00000

0.00000 0.20000 -0.20000 0.00000

0.00000 0.00000 0.00000 0.00000

>> G=B1'

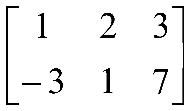
G =

0.00000 0.00000 0.00000 0.00000

0.00000 -0.10000 0.20000 0.00000

0.00000 0.60000 -0.20000 0.00000

0.00000 0.00000 0.00000 0.00000

Q3. A=  

A=[1 2 3;-3 1 7]

A =

1 2 3

-3 1 7

>> r=rank(A)

r = 2

>> B=[1 2;-3 1]

B =

1 2

-3 1

>> r1=rank(B)

r1 = 2

>>Binv=inv(B)

Binv =

0.14286 -0.28571

0.42857 0.14286

>>BinvT=Binv'

BinvT =

0.14286 0.42857

-0.28571 0.14286

A1=[0.14 0.42 0;-0.28 0.14 0]

A1 =

0.14000 0.42000 0.00000

-0.28000 0.14000 0.00000

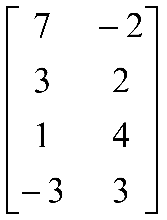
>> G=A1'

G =

0.14000 -0.28000

0.42000 0.14000

0.00000 0.00000

Q4. A = 

A=[7 -2;3 2;1 4;-3 3]

A =

7 -2

3 2

1 4

-3 3

>> r=rank(A)

r = 2

>> B=[7 -2;3 2]

B =

7 -2

3 2

>> r1=rank(B)

r1 = 2

>>Binv=inv(B)

Binv =

0.10000 0.10000

-0.15000 0.35000

>>BinvT=Binv'

BinvT =

0.10000 -0.15000

0.10000 0.35000

A1=[0 0;0.10 -0.15;0.10 0.35;0 0]

A1 =

0.00000 0.00000

0.10000 -0.15000

0.10000 0.35000

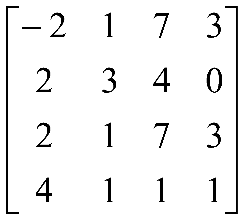
0.00000 0.00000

>> G=A1'

G =

0.00000 0.10000 0.10000 0.00000

0.00000 -0.15000 0.35000 0.00000

Q5. A=  

>> A=[-2 1 7 3;2 3 4 0;2 1 7 3;4 1 1 1]

A =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> rank(A)

Array indices must be positive integers or logical values.

'rank' appears to be both a function and a variable. If this is unintentional, use 'clear rank' to remove the

variable 'rank' from the workspace.

>> clear rank

>> rank(A)

ans =

4

>> sA=A

sA =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> Ait=inv(sA)'

Ait =

-0.2500 0.6000 -0.3250 0.7250

0.0000 0.2000 0.1000 -0.3000

0.2500 -0.8000 0.4750 -0.6750

-0.0000 0.6000 -0.4500 0.8500

>> G=Ait'

G =

-0.2500 0.0000 0.2500 -0.0000

0.6000 0.2000 -0.8000 0.6000

-0.3250 0.1000 0.4750 -0.4500

0.7250 -0.3000 -0.6750 0.8500

>> % to verify G is generalised inverse of A check two properties

%1. A\*G\*A=A

%2. G\*A\*G=G

A\*G\*A

ans =

-2.0000 1.0000 7.0000 3.0000

2.0000 3.0000 4.0000 0.0000

2.0000 1.0000 7.0000 3.0000

4.0000 1.0000 1.0000 1.0000

>> G\*A\*G

ans =

-0.2500 0.0000 0.2500 -0.0000

0.6000 0.2000 -0.8000 0.6000

-0.3250 0.1000 0.4750 -0.4500

0.7250 -0.3000 -0.6750 0.8500

>>

>> %%%%%%%%%% MPg-inverse of A

r=rref(A)

r =

1 0 0 0

0 1 0 0

0 0 1 0

0 0 0 1

>> C=A

C =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> R=r

R =

1 0 0 0

0 1 0 0

0 0 1 0

0 0 0 1

>> % check whether A=C\*R

C\*R

ans =

-2 1 7 3

2 3 4 0

2 1 7 3

4 1 1 1

>> % MPg-inverse of A is given by MPg=R'\*inv(C'\*A\*R')\*C'

MPg=R'\*inv(C'\*A\*R')\*C'

MPg =

-0.2500 0.0000 0.2500 -0.0000

0.6000 0.2000 -0.8000 0.6000

-0.3250 0.1000 0.4750 -0.4500

0.7250 -0.3000 -0.6750 0.8500

>> % to verify MPg by direct command

MPg=pinv(A)

MPg =

-0.2500 0.0000 0.2500 0.0000

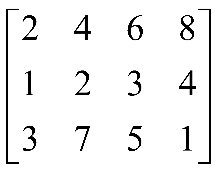
0.6000 0.2000 -0.8000 0.6000

-0.3250 0.1000 0.4750 -0.4500

0.7250 -0.3000 -0.6750 0.8500

>>%%% rank(A)=order(A) and A is square matrix hence there is no MPg-inverse

>>%% therefore MPg-inverse = g-inverse = conventional inverse

**Q6. D=**

D=[2 4 6 8;1 2 3 4;3 7 5 1]

D =

2 4 6 8

1 2 3 4

3 7 5 1

>> r=rank(D)

r = 2

>> E=[3 4;5 1]

E =

3 4

5 1

>> r1=rank(E)

r1 = 2

>>Einv=inv(E)

Einv =

-0.058824 0.235294

0.294118 -0.176471

>>EinvT=Einv'

EinvT =

-0.058824 0.294118

0.235294 -0.176471

D1=[0 0 0 0;0 0 -0.05 0.23;0 0 0.23 -0.17]

D1 =

0.00000 0.00000 0.00000 0.00000

0.00000 0.00000 -0.05000 0.23000

0.00000 0.00000 0.23000 -0.17000

>> G=D1'

G =

0.00000 0.00000 0.00000

0.00000 0.00000 0.00000

0.00000 -0.05000 0.23000

0.00000 0.23000 -0.17000

**Que2.**

**Obtain the inverse of matrix B by Partitioning method**

**To find the inverse of matrix B by Partitioning method**

B= [1 3 5 7;2 4 2 4;0 0 1 2;0 0 2 3]

B =

1 3 5 7

2 4 2 4

0 0 1 2

0 0 2 3

>> P=[1 3;2 4]

P =

1 3

2 4

>> p=inv(P)

p =

-2.0000 1.5000

1.0000 -0.5000

>> Q=[5 7;2 4]

Q =

5 7

2 4

>> R=[0 0; 0 0]

R =

0 0

0 0

>> S=[1 2;2 3]

S =

1 2

2 3

>> T=R\*p

T =

0 0

0 0

>> W=inv(S-T\*Q)

W =

-3.0000 2.0000

2.0000 -1.0000

>> Z=-1\*W\*T

Z =

0 0

0 0

>> Y=-1\*p\*Q\*W

Y =

-5.0000 6.0000

2.0000 -3.0000

>> X=p-Y\*T

X =

-2.0000 1.5000

1.0000 -0.5000

>> BINV=[X Y;Z W]

BINV =

-2.0000 1.5000 -5.0000 6.0000

1.0000 -0.5000 2.0000 -3.0000

0 0 -3.0000 2.0000

0 0 2.0000 -1.0000



**To find the inverse of matrix B by Partitioning method**

B=[1 0 0 0;3 -1 0 0;1 0 -1 0;0 1 -3 1]

B =

1 0 0 0

3 -1 0 0

1 0 -1 0

0 1 -3 1

>> P=[1 0;3 -1]

P =

1 0

3 -1

>> p=inv(P)

p =

1 0

3 -1

>> Q=[0 0;0 0]

Q =

0 0

0 0

>> R=[1 0;0 1]

R =

1 0

0 1

>> S=[-1 0;-3 1]

S =

-1 0

-3 1

>> T=R\*p

T =

1 0

3 -1

>> W=inv(S-T\*Q)

W =

-1 0

-3 1

>> Z=-1\*W\*T

Z =

1 0

0 1

>> Y=-1\*p\*Q\*W

Y =

0 0

0 0

>> X=p-Y\*T

X =

1 0

3 -1

>> BINV = [X Y;Z W]

BINV =

1 0 0 0

3 -1 0 0

1 0 -1 0

0 1 -3 1

**To find the inverse of matrix B by Partitioning method**

B=[-1 -3 0 1;3 5 8 -3;-2 -6 3 2;0 -1 2 1]

B =

-1 -3 0 1

3 5 8 -3

-2 -6 3 2

0 -1 2 1

>> P=[-1 -3;3 5]

P =

-1 -3

3 5

>> p= inv(P)

p =

1.2500 0.7500

-0.7500 -0.2500

>> Q=[0 1;8 -3]

Q =

0 1

8 -3

>> R=[-2 -6;0 -1]

R =

-2 -6

0 -1

>> S=[3 2;2 1]

S =

3 2

2 1

>> T=R\*p

T =

2.0000 -0.0000

0.7500 0.2500

>> W=inv(S-T\*Q)

W =

0.3333 0

-0.0000 1.0000

>> Z=-1\*W\*T

Z =

-0.6667 0.0000

-0.7500 -0.2500

>> Y=-1\*p\*Q\*W

Y =

-2.0000 1.0000

0.6667 0.0000

>> X=p-Y\*T

X =

4.5000 0.5000

-2.0833 -0.2500

>> BINV = [X Y;Z W]

BINV =

4.5000 0.5000 -2.0000 1.0000

-2.0833 -0.2500 0.6667 0.0000

-0.6667 0.0000 0.3333 0

-0.7500 -0.2500 -0.0000 1.00009